Problem 1
Write a complete C++ program that asks the user to enter a positive integer that leaves a remainder of either 3 or 4 when it is divided by 7. If the user gives incorrect input, the program should ask the user to try again as often as necessary. When the user succeeds, the program should report how many attempts were needed.
Sample output might be:

Enter a positive integer that is 3 or 4 modulo 7:    -3
Wrong. Try again: 9
Wrong. Try again: 10
Good. You passed after 3 attempts.

Problem 2
Write a complete C++ program that asks the user to enter a positive integer x. If the user enters a non-positive number the program should ask the user to try again as often as necessary. After the user has entered a positive value, the program should find (but not print) the remainders when x is divided by 7 and 8. It should then print the larger of these two remainders. (In case the two remainders are equal either can be printed.)
Sample output might be:

Enter a positive number: -4
Not positive. Try again: 0
Not positive. Try again: -2
Not positive. Try again: 20
Bigger remainder is:  6

Problem 3
Write a complete C++ program that does the following:
1. It asks the user to enter an integer between 100 and 9999.
2. If the entered number is out of range, the program forces the user to enter more numbers until one in the correct range is given.
3. Then the program prints the digits in the number (in reverse) on separate lines.
Here is an example of how the program should work:

Enter an integer between 100 and 9999:   8976
   6
   7
   9
   8

Problem 4
Write a complete C++ program that does the following. (Programs that correctly carry out some of the tasks will receive partial credit.)
1. It asks the user to enter a positive integer n.
2. It terminates when given illegal input.
3. It prints out the sum of those digits of n that are even numbers.
Here is an example of how the program should work:

Give me a positive integer:   41815

12

Problem 5
Write a complete C++ program that does the following. (Programs that correctly carry out some of the tasks will receive partial credit.)
1. It asks the user to enter a positive integer.
2. If the input is illegal, the program should terminate.
3. The program prints the digits of the number in reverse order (separated by spaces) and then gives their sum.
For example, if the user enters 19683 the program should print the following output.

3 8 6 9 1 sum to 27
Problem 6 Consider the following C++ program. What is the output from the program in response to the following user input?

```cpp
#include <iostream>
using namespace std;

int main() {
  int x, y;
  cout << "Please enter two positive integers: ";
  cin >> x >> y;
  if (x <= 0) {
    cout << "Illegal" << endl;
    exit (1);
  }
  if (y <= 0)
    cout << "Are you positive?\n";
  while (y < 10) {
    cout << y;
    y = y + x;
  }
  cout << y << endl;
  return 0;
}
```

(a) The user enters: -5 5
(b) The user enters: 5 -5
(c) The user enters: 10 1
(d) The user enters: 1 10
(e) The user enters: 1 1